Homework 3

In this assignment, you will experiment with text classification. You can continue using the movie review data:

https://github.com/dennybritz/cnn-text-classification-tf/tree/master/data/rt-polaritydata

Alternatively, you can pick any other dataset for text classification. Here's one interesting dataset:

https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge/data

There are a few similar ones here:

https://hatespeechdata.com

https://github.com/t-davidson/hate-speech-and-offensive-language

You will need to implement, i.e. write **from scratch**, your own vectorizer to convert the text of the input documents into vectors that can be used to train a scikit-learn classifier.

Please begin by studying the scikit-learn text classification tutorial:

https://scikit-learn.org/stable/tutorial/text_analytics/working_with_text_data.html

Once you are done, here are the concrete steps that you need to take (each is worth 20 points):

1. Randomly split the data into training (70)%, development (15%) and test (15%) sets. Keep the test set away in a separate location until you are done with model selection.
2. Implement your own vectorizer that converts the training data into a numpy array of shape (num_of_training_examples x num_of_features). To control the dimensionality of the resulting vectors, you can discard the features that occur in fewer than n examples and/or more than m examples. You will pick the actual value of n (e.g. 5) and/or m (e.g. 5000) in the process of model selection. Note: you are **not** allowed at this point to use third-party implementations such as scikit-learn's CountVectorizer etc.
3. Train a scikit-learn classifier of your choice (e.g. logistic regression or SVM) and use the development set to select the hyper-parameters of the classifier. Report the performance of your model on the development set.
4. Use the vectorizer you created in step 2 to convert the test set into a numpy array of shape: (num_of_test_examples x num_of_features). Use the best model from step 3 and report its performance on the test set.

5. Use scikit-learn CountVectorizer and/or TfIdfVectorizer to generate the input vectors, select the best hyper-parameters using the development set, and finally evaluate your best model using the test set. Compare your results with the results that you obtained with your manually-written vectorizer.

The ability to summarize your findings is extremely important when doing scientific research or working as a data scientist. Please summarize your findings in a 1-2 page paper. Include the details on what kind of pre-processing you performed and your choice of model hyper-parameters. This assignments will be evaluated primarily based on your writeup. Please see the general guidelines for homework submission in the syllabus.